An Analysis of Multi-Point Crossover

William M. Spears
Naval Research Laboratory
Washington, D.C. 20375 USA
spears@aic.nrl.navy.mil

Kenneth A. De Jong
George Mason University
Fairfax, VA 22030 USA
kdejong@aic.gmu.edu

Abstract

In this paper we present some theoretical results on two forms of multi-point crossover: n-point crossover and uniform crossover. This analysis extends the work from De Jong's thesis, which dealt with disruption of n-point crossover on 2nd order hyperplanes. We present various extensions to this theory, including 1) an analysis of the disruption of n-point crossover on kth order hyperplanes; 2) the computation of tighter bounds on the disruption caused by n-point crossover, by handling cases where parents share critical allele values; and 3) an analysis of the disruption caused by uniform crossover on kth order hyperplanes. The implications of these results on implementation issues and performance are discussed, and several directions for further research are suggested.

Keywords: Genetic algorithm theory, recombination operators

1 Introduction

One of the unique aspects of the work involving genetic algorithms (GAs) is the important role that recombination plays in the design and implementation of robust adaptive systems. In most GAs, individuals are represented by fixed-length strings and recombination is implemented by means of a crossover operator which operates on pairs of individuals (parents) to produce new strings (offspring) by exchanging segments from the parents' strings. Traditionally, the number of crossover points (which determines how many segments are exchanged) has been fixed at a very low constant value of 1 or 2. Support for this decision came from early work of both a theoretical and empirical

nature [Holland75, DeJong75].

However, there continue to be indications of an empirical nature that there are situations in which having a higher number of crossover points is beneficial [Syswerda89, Eschelman89]. Perhaps the most surprising result (from a traditional perspective) is the effectiveness on some problems of uniform crossover, an operator which produces on the average $(L / 2)$ crossings on strings of length L [Syswerda89].

The motivation for this paper is to extend the theoretical analysis of the crossover operator to include the multi-point variations and provide a better understanding of when and how to exploit their power. Specifically, this paper will focus on two forms of multi-point crossover: n-point crossover and uniform crossover.

2   Traditional Analysis

Holland provided the initial formal analysis of the behavior of GAs by characterizing how they biased the makeup of new offspring in response to feedback on the fitness of previously generated individuals. By focusing on hyperplane subspaces of L-dimensional spaces (i.e., subspaces characterized by hyperplanes of the form "---a----b---c--"), Holland showed that the expected number of samples (individuals) allocated to a particular kth order hyperplane H_k at time $t + 1$ is given by:

$$m(H_k, t+1) \geq m(H_k, t) * \frac{f(H_k)}{\bar{f}} * (1 - P_m k - P_c P_d(H_k))$$

In this expression, $f(H_k)$ is the average fitness of the current samples allocated to H_k, $\bar{f}$ is the average fitness of the current population, P_m is the probability of using the mutation operator, P_c is the probability of using the crossover operator, and $P_d(H_k)$ is the probability that the crossover operator will be "disruptive" in the sense that the children produced will not be members of the same subspace as their parents.

The usual interpretation of this result is that subspaces with higher than average payoffs will be allocated exponentially more trials over time, while those subspaces with below average payoffs will be allocated exponentially fewer trials. This assumes that there are enough samples to provide reliable estimates of hyperplane fitness, and that the effects of crossover and mutation are not too disruptive. Since mutation is typically run at a very low rate (e.g., $P_m = 0.001$), it is generally ignored as a significant source of disruption. However, crossover is usually applied at a very high rate (e.g., $P_c \geq 0.6$). So, considerable attention has been given to estimating P_d, the probability that a particular application of crossover will be disruptive.

To simplify and clarify the analysis, it is generally assumed that individuals are represented by fixed-length binary strings of length L, and that crossover points can occur with equal probability between any two adjacent bits. For ease of presentation these same assumptions will be made for the remainder of this paper. Generalizing the results to non-binary fixed-length strings is quite straightforward. Relaxing the other assumptions is more difficult.

Under these assumptions, Holland provided a simple and intuitive analysis of the disruption of 1-point crossover: as long as the crossover point does not occur within the defining boundaries of H_k (i.e., in between any of the k fixed defining positions), the children produced from parents in H_k will also reside in H_k [Holland75]. Figure 1 represents this graphically for a 3rd order hyperplane. Note that d_1, d_2, and d_3 represent the 3 defining positions of the 3rd order hyperplane, while P1 and P2 indicate the two parents.
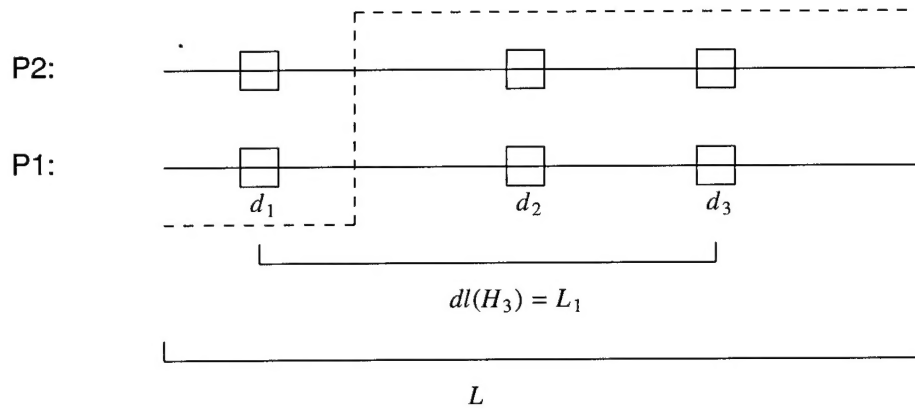
Figure 1: A 3rd Order Hyperplane

If crossover does occur inside the defining boundaries, disruption may or may not result. Disruption will depend on where the crossover point occurs inside the defining boundaries and on the alleles that the parents have in common on the k defining positions. Hence, P_d can be bounded by the probability that the crossover point will fall within the defining boundaries of H_k. Under the assumption of uniformly distributed crossover points, this yields:

$$P_d(H_k) \leq \frac{dl(H_k)}{L-1}$$

where $dl(H_k)$ is the "defining length" of H_k, namely the distance between the first and last of the k fixed defining positions of hyperplane H_k.

This analysis has lead to considerable discussion of the "representational bias" built into 1-point crossover, namely that crossover is much more disruptive to hyperplanes whose defining positions happen to be far apart. It also suggests a plausible role for inversion operators capable of effecting a change of representation in which the defining lengths of key hyperplanes are shortened.

De Jong [DeJong75] extended this analysis to n-point crossover by noting that no disruption can occur if there are an even number of crossover points (including 0) between each of the defining positions of a hyperplane. Hence, we have a bound for the disruption of n-point crossover:

$$P_d(n, H_k) \leq 1 - P_{k,even}(n, H_k)$$

where $P_{k,even}(n, H_k)$ is defined to be the probability that an even number of the n crossover points will fall between each of the k defining positions of hyperplane H_k. De Jong [DeJong75] provided an exact expression for $P_{k,even}$ for the special case of 2nd order hyperplanes (i.e., $k = 2$):

$$P_{2,even}(n, L, L_1) = \sum_{i=0}^{\left\lfloor \frac{n}{2} \right\rfloor} \binom{n}{2i} \left[\frac{L_1}{L} \right]^{2i} \left[\frac{L - L_1}{L} \right]^{n-2i}$$

$P_{2,even}(n, L, L_1)$ is the probability that an even number of crossover points will fall within the 2nd order hyperplane defined by L and L_1. Recall that L is the length of the string, while L_1 is the defining length of the hyperplane. The second term of the summation is the probability of placing an even number of crossover points within the 2 defining points. The third term is the probability of placing the remaining crossover points outside the 2 defining points. Finally, the combinatorial term $\binom{n}{2i}$ represents the number of ways an even number of points can be drawn from the n crossover points.

The family of curves generated by $P_{2,even}$ provide considerable insight into the change in disruptive effects on second order hyperplanes as the number of crossover points is increased. Figure 2 plots the curves for binary strings of length L. Notice how the curves fall into two distinct families depending on whether the number of crossover points is even or odd. Since $P_{2,even}$ guarantees no disruption, we're interested in increasing $P_{2,even}$ whenever possible. By going to an even number of crossover points, we can reduce the representational bias of crossover, but only at the expense of increasing the disruption of the shorter definition length hyperplanes.
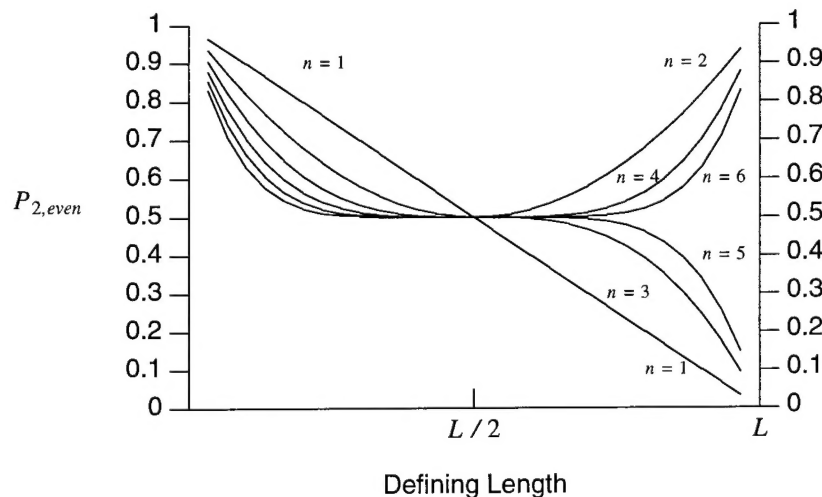

Figure 2. *n*-point Crossover Disruption on 2nd Order Hyperplanes

If we interpret the area above a particular curve as measure of the cumulative disruption potential of its associated crossover operator, then these curves suggest that 2-point crossover is the best as far as minimizing disruption. These results together with early empirical studies were the basis for using 2-point crossover in many of the implemented systems. Since then, there have been several additional studies focusing on crossover.

Bridges and Goldberg [Bridges85] have extended Holland's analysis of 1-point crossover, deriving tighter bounds on the disruption by taking into account the properties of the second parent and gains in samples in H_k due to disruption elsewhere.

Syswerda [Syswerda89] introduced a "uniform" crossover operator in which P_0 specified the probability that the allele of any position in an offspring was determined by using the

allele of the first parent, and $1 - P_0$ the probability of using the allele of the second parent. He provided an initial analysis of the disruptive effects of uniform crossover for the case of $P_0 = 0.5$, and compared it with 1 and 2 point crossover. He presented some provocative results suggesting that, in spite of higher disruption properties, uniform crossover can exhibit better recombination behavior, which can improve empirical performance.

Eschelman, Caruana, and Schaffer [Eschelman89] analyze crossover operators in terms of "positional" and "distributional" biases, and present a set of empirical studies suggesting that no n-point, shuffle, or uniform crossover operator is universally better than the others.

These results and other empirical studies motivated us to attempt to clarify the effects of multi-point crossover by extending the current analysis. In this paper we will present the following extensions:

1) An analysis of the disruption of n-point crossover on kth order hyperplanes.

2) The computation of tighter bounds on the disruption caused by n-point crossover, by examining the cases in which parents share common alleles on the hyperplane defining positions.

3) An analysis of the disruption caused by uniform crossover on kth order hyperplanes.

3 Crossover Disruption for Higher Order Hyperplanes

One possible explanation for the conflicting results on the merits of having more crossover points is that De Jong's analysis for the special case of 2nd order hyperplanes simply does not extend to higher order hyperplanes. In this section we attempt to resolve this issue by generalizing De Jong's results to hyperplanes of arbitrary order.

As noted earlier, the disruption probability $P_d(n, H_k)$ of n-point crossover on a kth order hyperplane H_k can be conservatively bounded by $1 - P_{k,even}(n, H_k)$ where $P_{k,even}(n, H_k)$ is the probability that n-point crossover produces only an even number of crossover points between each of the defining positions of H_k.

De Jong's formula for calculating $P_{2,even}$ can be generalized by noting that $P_{k,even}$ can be defined recursively in terms of $P_{k-1,even}$. To see this, consider how $P_{3,even}$ can be calculated in terms of $P_{2,even}$. Figure 3 illustrates the approach graphically.

The probability of n-point crossover generating only an even number of crossover points between $both$ d_1—d_2 and d_2—d_3 can be calculated by counting the number of ways an even number of crossover points can fall in between d_1—d_3, and for each of these possibilities requiring an even number to fall in d_1—d_2 (a second order calculation involving L_1 and L_2). More formally, we have:

$$P_{3,even}(n, L, L_1, L_2) =$$

$$\sum_{i=0}^{\left\lfloor \frac{n}{2} \right\rfloor} \binom{n}{2i} \left[\frac{L_1}{L} \right]^{2i} \left[\frac{L - L_1}{L} \right]^{n-2i} P_{2,even}(2i, L_1, L_2)$$
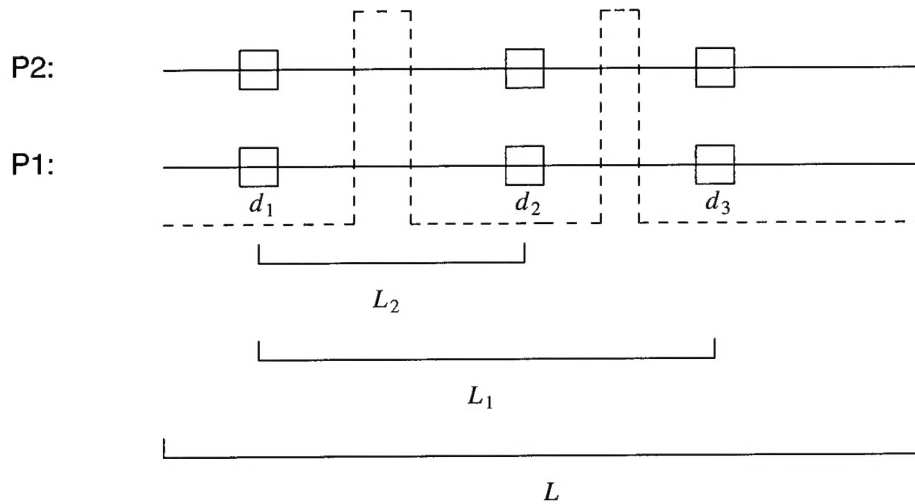
Figure 3. Non-disruptive n-point Crossover

In general, we have:

$$P_{k,even}(\, n, \, L, \, L_1, \ldots, \, L_{k-1} \,) =$$

$$\sum_{i=0}^{\lfloor \frac{n}{2} \rfloor} \binom{n}{2i} \left[\frac{L_1}{L} \right]^{2i} \left[\frac{L - L_1}{L} \right]^{n-2i} P_{k-1,even}(\, 2i, \, L_1, \ldots, \, L_{k-1} \,)$$

Figures 4 and 5 illustrate $P_{k,even}$ for hyperplanes of order 3 and 5. Each point on the graph represents an average over all hyperplanes of a particular defining length. Note that, apart from a skewing effect, the curves yield the same interpretation as De Jong's earlier curves for 2nd order hyperplanes: 2 point crossover minimizes disruption. So, extending the analysis thus far does not help in understanding the potential benefits of higher numbers of crossover points (seen in some empirical results).

4 Tighter Estimates on Disruption Probabilities

A second explanation for the conflicting results on the merits of a higher number of crossover points is that the $P_{k,even}$ curves are very weak bounds on P_d. It is possible that P_d itself, if analyzable, would yield different results. In this section we attempt to resolve this issue by providing tighter estimates on P_d.

The primary reason for the weakness of the $P_{k,even}$ bound is that it ignores the fact that many of the cases in which an odd number of crossover points fall between hyperplane defining positions are not disruptive to the sampling process. This occurs whenever the second parent happens to have identical alleles on the hyperplane defining positions which are exchanged by "odd" crossovers. (Note that an "odd" crossover occurs when an odd number of crossover points falls within 2 adjacent defining positions of the hyperplane.) Figure 6 illustrates this in the simple case of 2nd order hyperplanes. Note
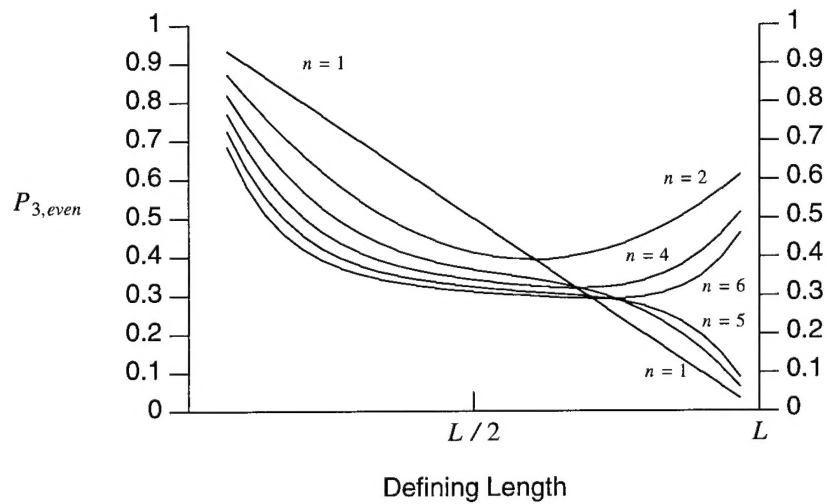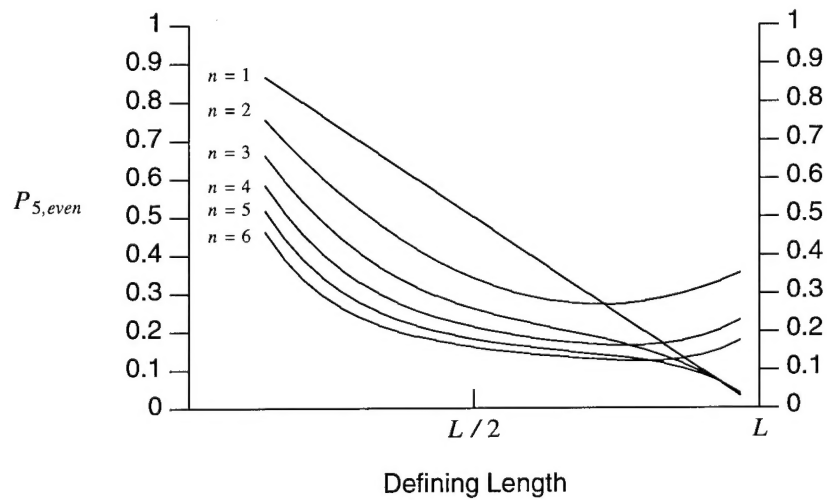
Figure 4. $P_{k,even}$ on 3rd Order Hyperplanes

Figure 5. $P_{k,even}$ on 5th Order Hyperplanes

that, in this figure, v_1 and v_2 represent the alleles (i.e., binary values) at those defining positions. Of the 4 possible combinations of matches on the defining positions of H_2, only the first ($-v_1\!\!-\!\!v_2\!-, -\bar{v}_1\!\!-\!\!\bar{v}_2\!-$) actually results in a disruption.

Deriving an expression for the probability that both parents will share common alleles on the defining positions of a particular hyperplane is difficult in general because of the
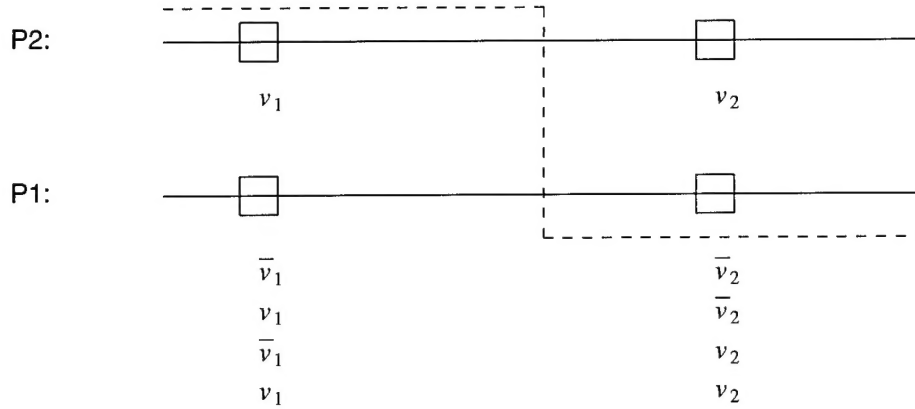
Figure 6. Disruption in "Odd" Crossovers

complexity of the population dynamics. We can, however, get a feeling for the effects of shared alleles on disruption by making the following simplifying assumption: the probability P_{eq} of two parents sharing an allele is constant across all loci.

With this assumption we can generalize $P_{k,even}$ to $P_{k,s}$ (i.e., the probability of *survival*) by including "odd" crossovers which are not disruptive. The generalization is still recursive in form:

$$P_{2,s}(n, L, L_1) = \sum_{i=0}^{n} \binom{n}{i} \left[\frac{L_1}{L} \right]^i \left[\frac{L-L_1}{L} \right]^{n-i} C$$

and

$$P_{k,s}(n, L, L_1, \ldots, L_{k-1}) =$$
$$\sum_{i=0}^{n} \binom{n}{i} \left[\frac{L_1}{L} \right]^i \left[\frac{L-L_1}{L} \right]^{n-i} P_{k-1,s}(i, L_1, \ldots, L_{k-1})$$

Notice that we are now summing over all crossover distributions (both even and odd), but have added a "correction" factor C at the "bottom" of the recursion to sort out the desired cases. C must be defined, then, for each path through the recursion. If each n is even at every level in that path, then there are an even number of crossover points between each of the defining positions. In this case, we define C to be 1, ensuring that all the even cases are counted as before. Suppose, however, that n is odd at some level in a path. Then there must be two adjacent defining positions that contain an odd number of crossover points. If C were defined to be 0 when this situation occurred, we would have exactly the same formulation as $P_{2,even}$ and $P_{k,even}$. However, we want to include those cases where the alleles of the parents on the hyperplane defining positions match in such a way that an "odd" crossover will not be disruptive. At the point where the recursion "bottoms out", a particular distribution of crossover points is completely specified. This, in turn enables one to identify how many of the given hyperplane's defining positions are being exchanged by this particular "odd" crossover. If both parents match on these positions, no disruption occurs.

As we saw in Figure 6, this will be the case for 2nd order hyperplanes if the parents match on either the first or the second or both defining positions. Hence, setting $C = P_{eq} + P_{eq} - (P_{eq})^2$ specifies the proportion of non-disruptive "odd" crossovers. If we assume that $P_{eq} = 0.5$ for example, then $C = 0.75$. This indicates that 75% of the "odd" crossovers are non-disruptive, which agrees with the prior discussion for Figure 6.

This same observation is true for kth order hyperplanes. If an "odd" crossover results in m of the k defining positions being exchanged, no disruption will occur if: 1) the parents match on all m positions being exchanged, or 2) if they match on all $k - m$ positions not being exchanged, or 3) they match on all k defining positions. Hence, the general form of the correction is:

$$C = P_{eq}{}^m + P_{eq}{}^{k-m} - P_{eq}{}^k$$

Figure 7 illustrates this for one particular "odd" crossover on 4th order hyperplanes.
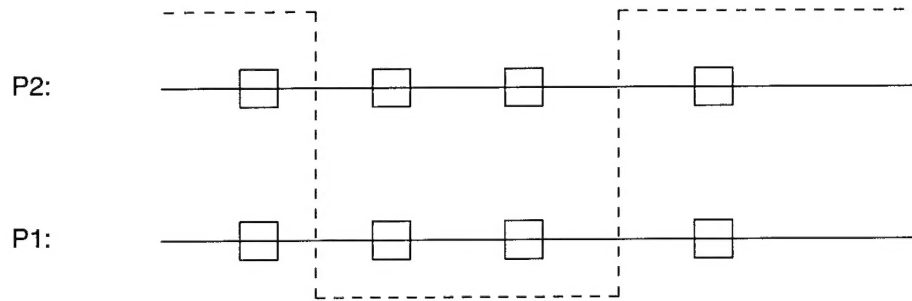

Figure 7. Non-disruptive "Odd" Crossover on 4th Order Hyperplanes

In this case,

$$C = P_{eq}{}^2 + P_{eq}{}^2 - P_{eq}{}^4$$

If $P_{eq} = 0.5$, then $C = (7 / 16)$ reflects the proportion of cases in which this particular crossover will not be disruptive.

Figures 8 and 9 show the effects of counting the non-disruptive "odd" crossovers. Figure 8 assumes a value of $P_{eq} = 0.5$, which is likely to hold in the early generations when matches are least likely. Figure 9 assumes a value of $P_{eq} = 0.75$ to get a feeling of the effect as the population becomes more homogeneous. Note that in both cases, the amount of expected disruption has been significantly reduced and the relative difference in disruption among different numbers of crossover points is reduced as well. At the same time, note that the curves for the various number of crossover points have held their relative position with respect to one another.

These results help explain the fact that in some empirical studies little or no difference in effect is seen by varying the number of crossover points between, say, 1 and 16. It does not appear to explain why in some situations more crossover points and, in particular, uniform crossover seems to perform significantly better.
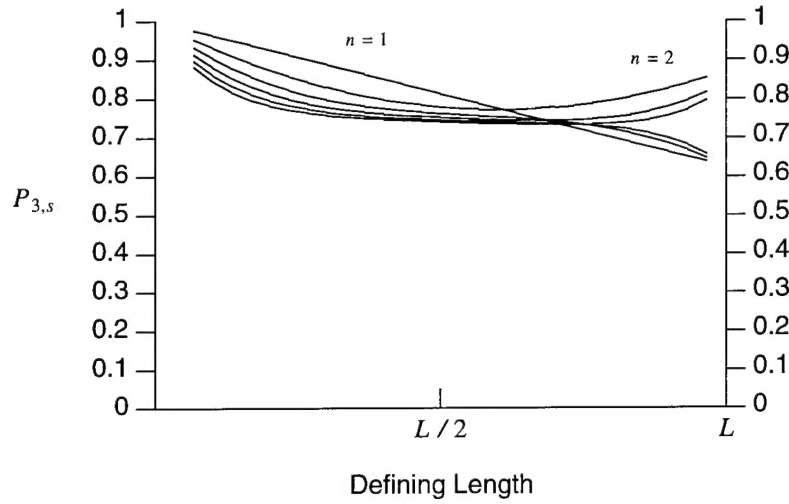
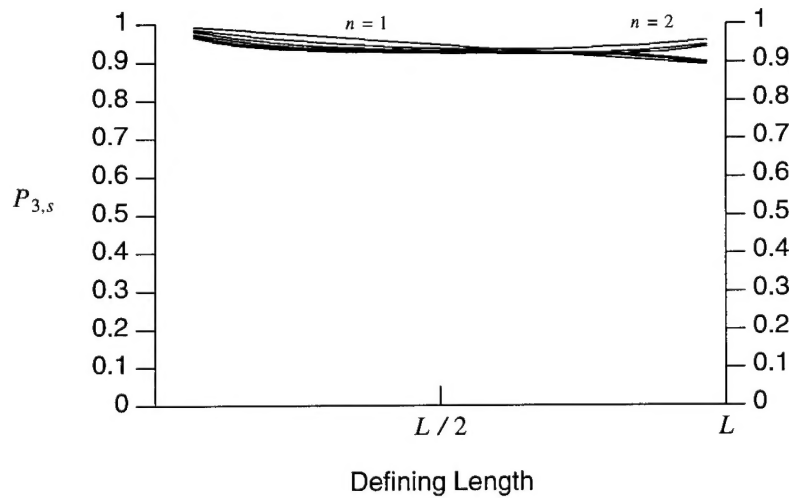Figure 8. $P_{k,s}$ on 3rd Order Hyperplanes with $P_{eq} = 0.5$

Figure 9. $P_{k,s}$ on 3rd Order Hyperplanes with $P_{eq} = 0.75$

5 Analyzing Uniform Crossover

Syswerda [Syswerda89] defined a family of "uniform" crossover operators which is a variant of a notion that has been informally experimented with in the past: to produce offspring by randomly selecting at each loci the allele of one of the parents. By defining P_0 to be the probability of using the first parent's allele, offspring can be produced by

flipping a P_0 biased coin at each position. (Other informal studies viewed the process as a random walk and defined P_0 as the probability of switching over to the other parent. The two views are equivalent if and only if $P_0 = 0.5$.)

A good way of relating uniform crossover to the more traditional n-point crossover is to think of uniform crossover as generating a mask of 0s and 1s, indicating which parent's allele is to be used at each position. As we scan the mask from left to right, a switch from 0 to 1 or from 1 to 0 represents a crossover point. For example, the mask 0011100 defines a 2-point crossover operation. If $P_0 = 0.5$, all masks are equally likely. If we examine the n-point crossover operations defined by this set of masks, we see immediately that they are binomially distributed around $((L-1)/2)$. For example, the set of all 4-bit masks defines:

<div align="center">

2 0-point crosses
6 1-point crosses
6 2-point crosses
2 3-point crosses

</div>

If $P_0 \neq 0.5$, the masks are no longer uniformly distributed, but contain on the average longer runs of 0s or 1s. From the point of view of n-point crossover, the effect is to skew the binomial distribution toward 0.

We are now in a position to analyze the disruption properties of uniform crossover in the same manner as the analysis of n-point crossover in the preceding sections. We note that the notion of an even number of crossover points between the defining positions of hyperplane H_k corresponds to masks which have either all 0s or all 1s on the defining positions of H_k. Hence, the corresponding conservative bound on the disruption of uniform crossover is given by:

$$P_d(H_k) \leq 1 - P_{k,even}(H_k)$$

where

$$P_{k,even}(H_k) = (P_0)^k + (1-P_0)^k$$

If $P_0 = 0.5$ for example, then

$$P_{k,even}(H_k) = (\frac{1}{2})^{k-1}$$

for all hyperplanes of order k. Notice that, unlike the traditional n-point crossover, there is no representational bias with uniform crossover in the sense that all hyperplanes of order k are equally disrupted regardless of how long or short their defining lengths are.

As before, we can get a tighter estimate of P_d if we include non-disruptive "odd" crossovers. For uniform crossover this corresponds to those masks which are not either all 0s or all 1s on the hyperplane defining positions, but are non-disruptive because the parents share common alleles on those particular positions. More formally, we have

$$P_{k,s}(H_k) = P_{k,even}(H_k) + \sum_{i=1}^{k-1} \begin{bmatrix} k \\ i \end{bmatrix} (P_0)^i (1-P_0)^{k-i} (P_{eq}^{\ i} + P_{eq}^{\ k-i} - P_{eq}^{\ k})$$

where P_{eq} is the probability of matching alleles, as before. Note that the last term in the expression is identical to the correction C defined earlier for the n-point crossover analysis. If the above is rewritten more concisely, $P_{k,s}$ can be expressed in a form similar to that derived for the n-point analysis:

$$P_{k,s}(H_k) = \sum_{i=0}^{k} \binom{k}{i} (P_0)^i (1 - P_0)^{k-i} (P_{eq}^{i} + P_{eq}^{k-i} - P_{eq}^{k})$$

Figure 10 illustrates the relationship between uniform crossover and n-point crossover for 3rd order hyperplanes. Note that, as expected, uniform crossover does not minimize disruption but, at the cost of higher disruption, removes any representational bias. This helps to explain why uniform crossover can yield performance improvements in some cases. Consider situations in which the critical low order hyperplanes happen to be widely separated in a particular representation. Uniform crossover significantly reduces the disruption pressure on these critical hyperplanes at the expense of more disruption on the adjacent (but non-critical) low order hyperplanes. However, in the reverse situations in which the representation happens to place critical positions close together, 1 and 2 point crossover is more effective.
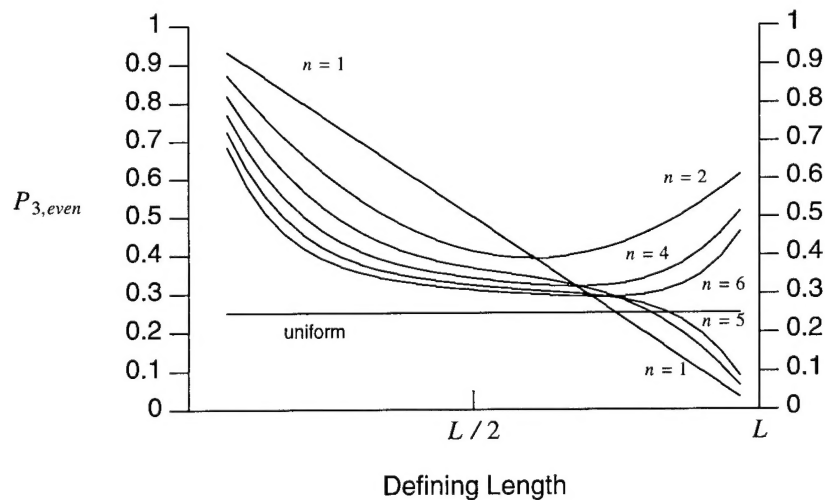
Figure 10. Disruption of Uniform Crossover

6   Is Disruption Always Bad?

So far, the analysis of crossover has focused on its potential for sampling disruption with the implication that disruption is bad. Sampling disruption is important for understanding the effects of crossover when populations are diverse (typically early in the evolutionary process). However, when a population becomes quite homogeneous, another factor becomes important: whether the offspring produced by crossover will be different than their parents in some way (thus generating a new sample) or just clones. This property of crossover has been dubbed "crossover productivity" and is easy to measure. Figure 11 illustrates how significantly the "productivity" of 2-point crossover can drop off as evolution proceeds. The horizontal axis indicates the number of generations the GA has run (i.e., we use a generational GA). The vertical axis indicates the number of crossovers, at each generation, that produced offspring different from their parents. Since $P_c = .6$, and the population size is 100, the maximum productivity is 60. The problem examined, HC11, is a boolean satisfiability problem explained in

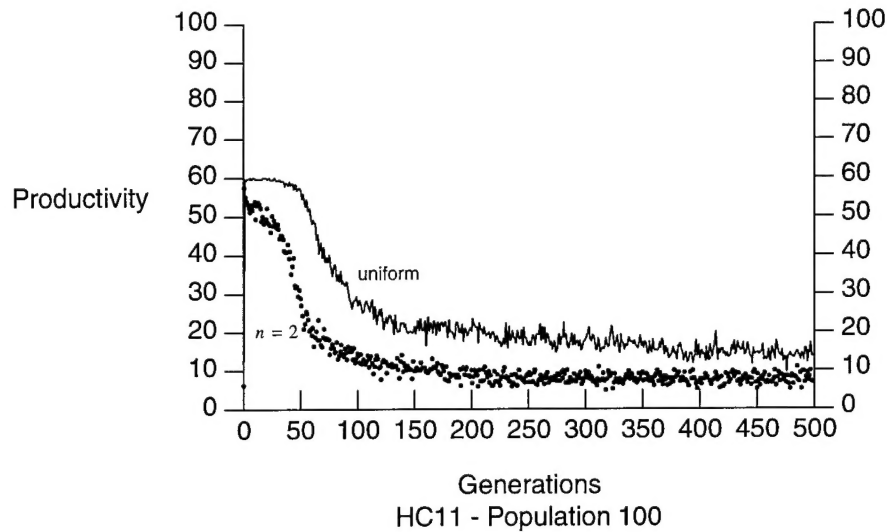[Spears90]. The problem has 55 binary variables, and has one unique solution with a fitness of 1.0.†

Figure 11. Productivity of 2-point & Uniform Crossover

If we try to formally compute the probability that the offspring will be different than their parents, the computation is precisely the same as the previous disruption computations. To see this, consider two parents whose alleles differ on only 4 loci. In order for crossover to produce new offspring, some but not all of those alleles must be exchanged. The probability of this occurring is just $P_d(H_4)$. In other words, those operators that are more disruptive are also more likely to create new individuals from parents with nearly identical genetic material.

This observation helps explain some of the other experimental results in which higher crossover rates performed better. Figure 12 is an example of one such result. Again, the horizontal axis represents generations. The vertical axis represents the best individual seen. Notice that 2-point crossover converges more quickly, but to a lower plateau than uniform crossover which converges more slowly to a better solution.

This suggests two additional directions for research. First, note that it may be possible to have the best of both worlds by modifying 2-point crossover to be less likely to produce clones. This can be achieved in a brute force way by repeated calls to crossover until non-clones are produced, or in more sophisticated ways such as Booker's reduced surrogate approach [Booker85]. Figure 13 illustrates the effect of the brute force technique on one particular example. Notice that this change has little effect during the early generations when children are most likely to be different anyway. However, the increased "productivity" in the later stages slows the early convergence seen before.

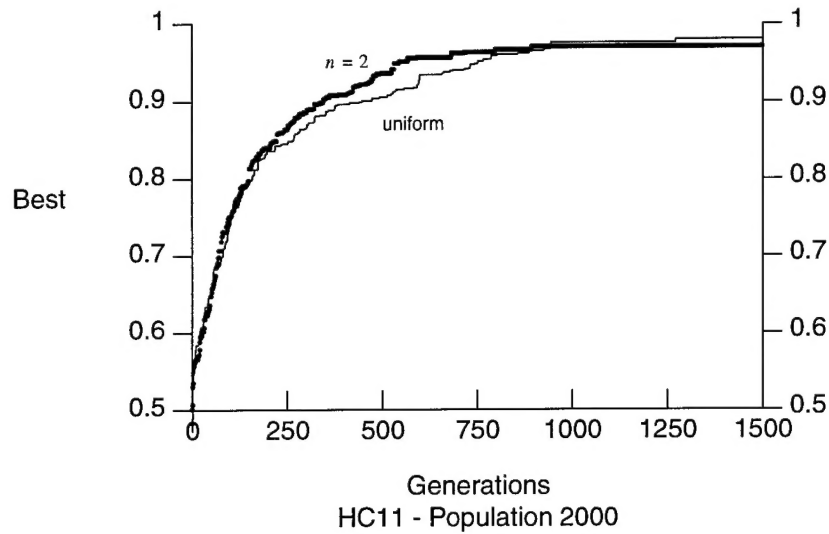† All experimental results are averaged over at least 10 independent runs.

Figure 12. Productivity-related Performance of 2-point & Uniform Crossover
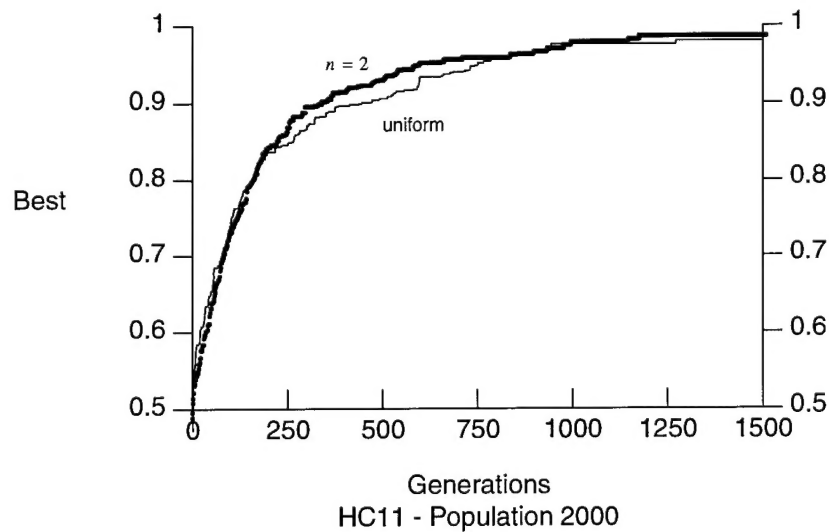

Figure 13. 2-point Crossover Augmented to Increase Productivity

The second direction for future research is the obvious interaction of multi-point crossover and population size. Smaller population sizes tend to converge faster to levels of homogeneity which reduce crossover productivity. With larger population sizes the effects appear to be much less dramatic. This suggests a way to understand the role of multi-point crossover. With small populations, more disruptive crossover operators such

as uniform or n-point ($n \gg 2$) may yield better results because they help overcome the limited information capacity of smaller populations and the tendency for more homogeneity. However, with larger populations, less disruptive crossover operators (2-point) are more likely to work better, as suggested by Holland's original analysis.

7 Conclusions and Further Work

The extensions to the analysis of n-point and uniform crossover presented in this paper provide additional insight into the role and effective use of these operators. At the same time, this analysis has suggested some directions for further research. The authors are currently involved in extending the results presented here to include the interacting effects of population size and crossover productivity. The view we are taking is that there is very little likelihood of finding globally correct answers to questions such as the choice of population size and crossover operators. Our goal is to understand these interactions well enough so that GAs can be designed to be self-selecting with respect to such decisions.

References

Booker, Lashon B. (1987). Improving Search in Genetic Algorithms, *Genetic Algorithms and Simulated Annealing*, Morgan Kaufmann Publishing.

Bridges, C. & Goldberg, D. (1985). An Analysis of Reproduction and Crossover in a Binary-Coded Genetic Algorithm, *Proc. 3rd Int'l Conference on Genetic Algorithms*, Lawrence Erlbaum Publishing.

De Jong, Kenneth A. (1975). *An Analysis of the Behavior of a Class of Genetic Adaptive Systems,* Doctoral thesis, Dept. Computer and Communication Sciences, University of Michigan, Ann Arbor.

Eschelman, L., Caruana, R. & Schaffer, D. (1989). Biases in the Crossover Landscape, *Proc. 3rd Int'l Conference on Genetic Algorithms*, Morgan Kaufman Publishing.

Goldberg, David E. (1989). *Genetic Algorithms in Search, Optimization & Machine Learning,* Addison-Wesley Publishing Company, Inc.

Holland, John H. (1975). *Adaptation in Natural and Artificial Systems,* The University of Michigan Press.

Spears, William M. (1990). *Using Neural Networks and Genetic Algorithms as Heuristics for NP-Complete Problems*, Masters Thesis, Department of Computer Science, George Mason University, Fairfax, Virginia, 1990.

Syswerda, Gilbert. (1989). Uniform Crossover in Genetic Algorithms, *Proc. 3rd Int'l Conference on Genetic Algorithms*, Morgan Kaufman Publishing.